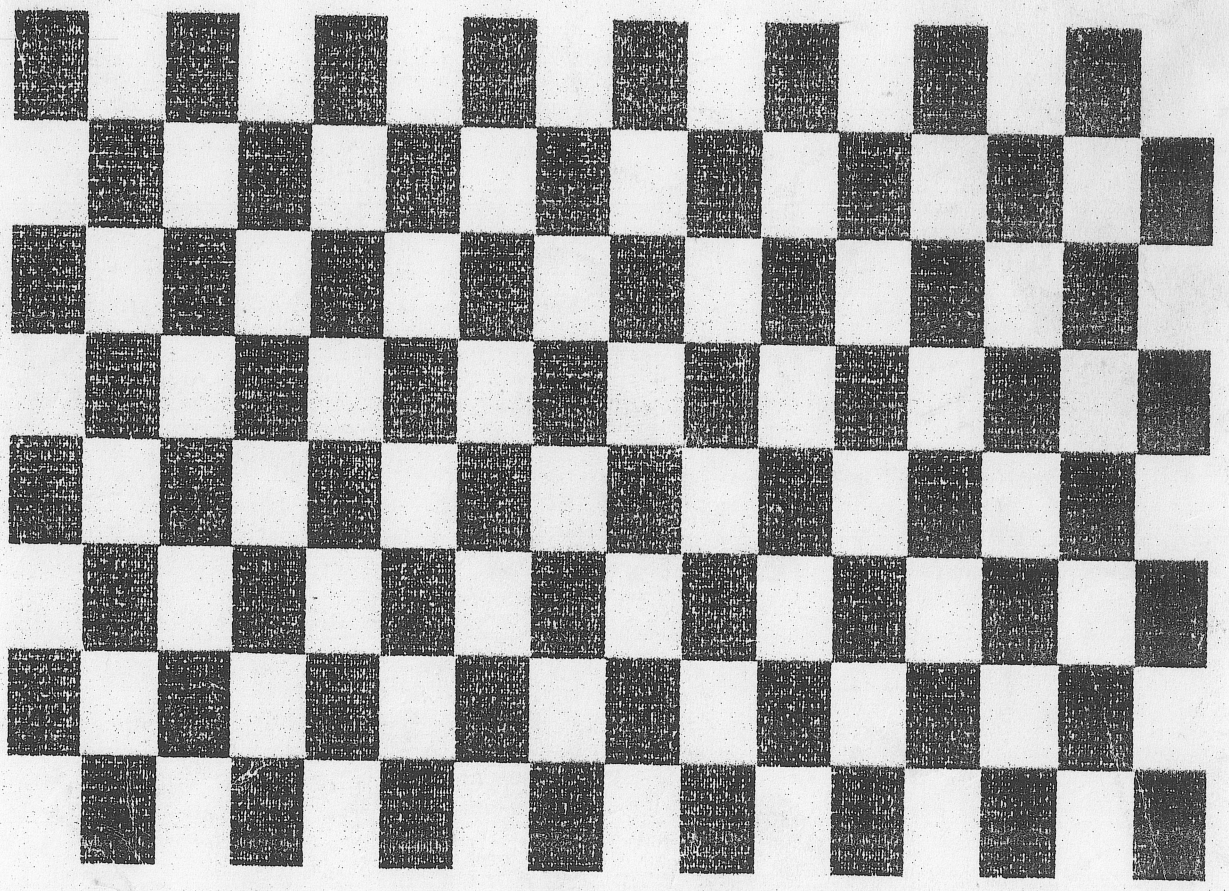
MARCH/APRIL 1989 ISSUE # 23

VZ 200/300

HUNTER VALLEY VZ JOURNAL



PRODUCED BI-MONTHLY BY H.V.VZ.U.G.

THIS PUBLICATION WAS PREPARED ON A STAR NX 1000 PRINTER USING DAVE MITCHELL'S PATCH3.3 TOGETHER WITH E & F WORD PROCESSOR. HI & LO-RES SCREEN DUMPS AND LISTINGS WERE DONE USING LARRY TAYLOR'S PRINTER PATCH V1.4 AVAILABLE FROM VSOFTWAREZ WHILE PATCH 3.3 IS AVAILABLE FROM HUNTER VALLEY VZ USERS' GROUP.

FRONT COVER - USUALLY I DON'T HAVE MUCH OF AN IDEA WHAT'S GOING ON THE COVER AND THIS ISSUE EVEN SURPRISED ME. SEE PAGE 13.

CLUB NEWS/NEXT MEETING/COMPUTER SHOW/ETC . . . . . PAGE 3

NUMBER SYSTEMS BY LARRY TAYLOR . . . . . PAGES 4-5
IF DECIMAL, BINARY & HEX LEAVE YOU A BIT CONFUSED THEN LARRY WILL CLEAR THINGS UP FOR YOU IN HIS EASY TO UNDERSTAND STYLE.

TAPE/DISK CONVERTER FOR SOURCE FILES BY BRIAN GREEVE .. PAGES 6-7
ONCE AGAIN BRIAN PRESENTS US WITH A USEFULL UTILITY WHICH WILL ALLOW YOU TO TRANSFER SOURCE CODE FILES FROM TAPE TO LASERLINK DISK BASED EDITOR ASSEBLER.

ENHANCING THE LIST COMMAND BY DAVE MITCHELL . . . . . PAGE 8
ANOTHER COUPLE CLEVER UTILITIES FROM DAVE WHICH WILL WHEN ENABLED LIST LINE NUMBER ADDRESSES IN DECIMAL AND IN HEX.

HI-RES SCREEN MOVE & PRESEVE UTILITIES BY BOB KITCH PAGES 9-11
THIS VERY WELL PRESENTED AND DOCUMENTED UTILITY IS DESIGNED TO MOVE ANY HI-RES SCREEN FROM VIDEO RAM TO TOM AND BACK AGAIN AND WOULD BE IDEAL FOR ANY GRAPHICS PROGRAM. YOU COULD TRY IT WITH ROBERT QUINN'S HI-RES DRAWER FROM LAST ISSUE.

HI/LO-RES COPIER PART II BY ROBERT QUINN . . . . . PAGES 12-13
ROBERT HAS INCORPORATED A BLOCK MOVE ROUTINE WHICH WILL MOVE ANY HI/LO-RES SCREEN BETWEEN VIDEO RAM AND HI-MEM. THIS OPTION ALLOWS TO RECALL A SCREEN FROM HI-MEM JUST BY A KEY PRESS OR TWO.

128K SIDEWAYS RAM PART III BY JOE LEON . . . . . PAGES 14-15
MODIFIED CIRCUIT IS GIVEN TO MAKE IT COMPATIBLE WITH 64K RAM PACK AND VZ 200 CONSTRUCTION IDEAS. MY APOLOGIES FOR ARTICLE AS IT'S NOT AS GOOD AS I WOULD HAVE LIKED IT TO BE.

VZ SUPER GRAPHICS PART II BY JOE LEON . . . . . PAGES 16-18
VZ 200 VEROBOARD LAYOUT IS GIVEN TOGETHER WITH PARTS LIST MISSED FROM LAST ISSUE. AGAIN MY APOLOGIES AS I FEEL I COULD HAVE PRESENTED ARTICLE MUCH BETTER. PROGRAMMING, DETAILED EXPLANATION, ETC HELD OVER FOR NEXT ISSE.

FOR PRIVATE SALE . . . . . PAGES 3, 19 & 20

VZ USER GROUPS - FOR SALE - SCREEN ED . . . . . PAGE 19

FOR SALE - NEW PATCH3.3 - EXT.DOS - MENU/FILE COPIER . . PAGE 20

CUSTOMIZING E & F WORD PROCESSOR PRINTER MENU WAS HELD OVER AND SHOULD APPEAR IN NEXT ISSUE.

COMING ISSUES :-

ATARI TYPE JOYSTICK ADAPTOR FOR VZ 200/300 - EPROM ERASER
128K S/WAYS RAM FOR 4000H-5FFFH RANGE (DOS AREA)
AUTO START/STOP FOR DATASSETTES
34K USER RAM + 2K RAM FOR 6000H RANGE FOR VZ200

APOLOGIES - AS A RESULT OF MY CAR ACCIDENT LAST YEAR I'VE BEEN LEFT WITH CONSTANT PAINS AND HEADACHES WHICH MY JOB AS TRUCK DRIVER MAKES WORSE. I FIND IT DIFFICULT TO CONCENTRATE AND I'VE SLOWED DOWN CONSIDERABLY WITH RESULT THAT QUITE A FEW LETTERS AND REQUESTS FOR HELP HAVE BEEN UNANSWERED, SOME AWAITING A REPLY FOR MONTHS.

PLEASE ACCEPT MY APOLOGIES AND I'LL TRY TO ANSWER YOUR LETTERS/REQUESTS AS SOON AS I CAN. TO MAKE MATTERS WORSE LOCAL COMPUTER SHOW IS ON NEXT WEEK AND THERE'S LOTS TO PREPARE BEFORE THEN AND AS A RESULT JOURNALS MAY BE POSTED OUT LATE AND I HOPE YOU'LL UNDERSTAND. ALSO I'LD LIKE TO THANK YOU ONCE AGAIN FOR YOUR CONTINUED WISHES FOR MY SPEEDY RECOVERY, MUCH APPRECIATED.

CONTRIBUTIONS TO THE JOURNAL :-

MY THANKS FOR YOUR CONTINUED SUPPORT VIA YOUR CONTRIBUTIONS AND THEIR EXELLENT LAYOUT AND CONTENT. FOR A CHANGE THERE ARE SOME LEFT OVER FOR NEXT ISSUE/S. PLEASE KEEP CONTRIBUTIONS COMING AS IT MAKES MY JOB LOTS EASIER AND IT MAKES FOR GREATER VARIETY.

IF YOU'RE THINKING OF CONTRIBUTING TO THE JOURNAL THE PREFERED FORMAT IS BASIC LISTINGS, WORD PROCESSOR AND SOURCE CODE FILES ON TAPE OR DISK. FILES FROM THE FOLOWING WORDPROCESSORS CAN BE ACCEPTED :-

E & F TAPE OR DISK, WORDPRO CARTRIDGE & ALL QUICKWRITE WORDPROCESSOR FILES. ALSO TAPE OR DISK EDITOR ASSEMBLER FILES.

HUNTER VALLEY VZ USERS' GROUP CLUB MEETINGS :-
APRIL 7 - MAY 5 - JUNE 2 (ANNUAL GENERAL MEETING)

IF YOU ONLY ATTEND ONCE A YEAR THEN PLEASE COME TO JUNE MEETING AS THE CLUBS FUTURE IS IN YOUR HANDS. IT CANNOT FUNCTION WITHOUT YOUR SUPPORT.

6TH ANNUAL NEWCASTLE MICROCOMPUTING EXHIBITION - APRIL 13-15

VENUE - HUNTER INSTITUTE OF HIGHER EDUCATION - WARATAH

THE HUNTER VALLEY VZ USERS' GROUP HAS BEEN INVITED TO PARTICIPITATE AGAIN. HOPE TO SEE YOU THERE, APRIL 14 & 15 ONLY.

WANTED TO BUY - VZ 200 6K RAM BOARDS WITH OR WITHOUT 2K RAM CHIPS. NEEDED FOR FUTURE PROJECTS. - PLEASE CONTACT JOE LEON IF YOU CAN HELP OUT.

DISCLAIMER - EVERY EFFORT IS MADE TO ENSURE THE ACCURACY OF INFORMATION CONTAINED WITHIN BE IT GENERAL, TECHNICAL, PROGRAMMING, ETC. NO RESPONSIBILITY CAN BE ACCEPTED BY H.V.VZ.U.G. OR AUTHOR AS A RESULT OF APPLYING SUCH INFORMATION.

THE HUNTER VALLEY VZ JOURNAL IS SUBJECT TO COPYRIGHT AND NO MATERIAL IN THIS JOURNAL MAY BE REPRODUCED IN PART OR WHOLE WITHOUT THE CONSENT OF THE AUTHOR WHO RETAINS COPYRIGHT.

MATHS MASTERY OR MYSTERY?

One of the problems associated with computers for newcomers is an encounter with number systems other than the one we use every day. The first thing to understand is that the numbers themselves don't change, only the way in which they are represented. The symbols used, when depicting numbers are called digits, a throwback to the days, when people used to count on their fingers and toes.

Number systems are frequently identified by the number of different digits they use. Our present, widely used number system uses ten digits, 0,1,2,3,4,5,6,7,8,9 and for this reason alone, it is called a decimal number system. "deci" comes from the Latin word for "ten". Because it is based on the number 10, it is also called a base 10 number system. Having only ten digits means that to write large numbers, the digits must be reused and this results in the idea of place value. That is, a digit has a value assigned to it according to the position it occupies in a number. In the base 10 (decimal) number system, as a digit moves each place to the left, its value increases ten times.

Place value	1000	100	10	1
	7	7	7	7

This number is seven thousand, seven hundred and seventy-seven.

The two number systems most frequently met, when dealing with computers, are binary and hexidecimal. The first (binary) represents numbers at the level, at which the computer deals with them, whilst the second (hexidecimal) is meant to be more convenient for us to use. The word "binary" contains the prefix "bi" meaning "two", and gets its name from the fact that it uses only two digits 0 and 1. This means that even relatively small numbers can appear quite large, when we use binary to represent them. Because binary is based on 2, it is sometimes called a base 2 number system. A digit in a binary number is still assigned a value according to its position, but in this case the value of a digit increases by only 2 times as it moves each place to the left.

Place value	8	4	2	1
	1	0	0	1

This number is read as one, zero, zero, one (base two).

(It is important to note, that 1001_2 is a binary number with a value equal to the base ten number 9 (8+1), and as such should *never* be read as one thousand and one.)

To the computer, the one and zero represent on-off states like in a switch. In an 8 bit computer, the biggest number that can be stored in a single memory location is 255. This is because an 8 bit number is one made up of 8 *bi*nary dig*its* or 8 *bits*. The biggest 8 bit number would have all eight digits as ones.

eg.

128	64	32	16	8	4	2	1
1	1	1	1	1	1	1	1

The place value of each digit is shown above it. By adding these values, (128+64+32+16+8+4+2+1), we arrive at a total of 255.

Hexidecimal, coming from "hex" meaning "six" and "deci" meaning "ten" stands for sixteen. As you might expect, given its name, the hexidecimal number system has sixteen different digits. Therefore, hexidecimal or hex, as it is frequently called, is referred to as a base 16 number system. The first ten digits, taken from the decimal number system, are the familiar 0,1,2,3,4,5,6,7,8,9, whilst the remaining six, having to be just as familiar are borrowed from our alphabet, namely the letters A,B,C,D,E and F. The value assigned to each digit in a hexidecimal number increases by 16 times as it moves each place to the left.

Place Value	4096	256	16	1
	2	4	8	6

This number is read as two, four, eight, six hex.

The main reason for using hexidecimal is that all numbers up to 255 (ie. all 8 bit numbers), can be represented by a 2 digit hex number from 00 to FF. Two byte (or 16 bit) numbers up to 65 535 can be represented by a 4 digit hex number from 0000 to FFFF. In addition, the first two digits in a 4 digit hex number represent the high (or most significant) byte, whilst the second pair of digits give the low (or least significant) byte. Such an arrangement is useful when displaying numbers on the screen, since all two byte numbers, irrespective of magnitude, occupy only four spaces. A brief comparison of how numbers are represented by the three number systems is given below.

Base 10	16	2	Base 10	16	2
1	01	00001	9	09	01001
2	02	00010	10	0A	01010
3	03	00011	11	0B	01011
4	04	00100	12	0C	01100
5	05	00101	13	0D	01101
6	06	00110	14	0E	01110
7	07	00111	15	0F	01111
8	08	01000	16	10	10000

Why bother with these other number systems? Hexidecimal is most often encountered when using disassembler or assembler programs, since screen formatting is made much simpler. Disassembling machine code programs is the means to understanding how they work and using assembly language is by far the easiest way to communicate directly with the CPU (Central Processing Unit). Without an understanding of hex, these doorways to further understanding will remain forever closed.

Binary is frequently encountered, when you begin to delve into the hardware side of the computer and start unravelling the logic circuits used. A knowledge of it is also useful, when examining the many single bit flags used in the communications area of RAM, and more importantly, when setting and resetting them by poking appropriate values into memory. The keyboard latch and the screen are both bit mapped and require some understanding of binary to enable the user to interpret them properly.

In short.....

If you pass binary by, with nary a thought.
Or let hex remain, a mysterious spell.
You'll face your machine, for years to come,
Exploring its secrets, but not too well.

1 X DISK DRIVE & DISK CONTROLLER - $280.00
(DISK CONTROLLER INCLUDES 2ND SOCKET WITH 8K RAM).
2 X ORIGINAL VZ TAPE DECKS - $50.00 EACH.
ORIGINAL SOFTWARE TAPES - MYSTERY PACKS - $ CALL
SOFTWARE ON DISKS - MYSTERY PACKS - 5 DISKS = $18.00
ALL PRICES INCLUDE P/P UP TO $10/$12.
1 X EXTENDED KEYBOARD (THE MICROBEE ONE) WIRED FOR VZ $40.00
(INCLUDES P/P).
1 X DUAL PRINTER BUFFER (2 X 64K UNITS IN BOX) - $400.00 -
(REQUIRES POWER SUPPLY) - EACH UNIT CAN BE UPGRADED TO 256K
INDEPENDENTLY OF THE OTHER.

FOR MORE INFO ON THIS UNIT OR ANY ABOVE PLEASE CONTACT :-
DAVE BOYCE 41 HEATHER DRIVE CHRISTIE DOWNS S.A. 5164
PHONE (08) 384 6574

TAPEDISK SOURCE CONVERTER by Brian GREEVE

THIS PROGRAM IS INTENDED FOR THOSE PROGRAMMERS WHO MAY HAVE COPIES OF SOURCE CODE ON TAPE AND WISH TO USE THEM WITH THE LASERLINK DISK BASED EDITOR ASSEMBLER.

TO USE, THE EDITOR ASSEMBLER IS BRUN AND THE EXIT TO BASIC IS MADE (TB). BRUN "TAPEDISK" AND WAITING WILL APPEAR IN THE LOWER LEFT HAND CORNER. THIS PROGRAM WILL LOAD THE FIRST SOURCE CODE FILE IT COMES TO. ALL OTHER FILES ARE IGNORED.

FOR SIMPLICITY THERE IS NO FACILITY TO SPECIFY THE DESIRED FILE NAME BUT THIS IS IN LINE WITH THE ORIGINAL TAPE BASED EDITOR ASSEMBLER PROGRAM. WHEN THE FILE IS LOADED THE PROGRAM JUMPS BACK TO THE EDITOR ASSEMBLER WHERE THE FILE MAY BE USED, AMENDED, OR SAVED TO DISK.

THE PROGRAM IS OFFERED IN THE FORM OF A BASIC LOADER AND AS A SOURCE CODE LISTING. BEFORE YOU CAN TRANSFER YOUR SOURCE CODE FILES FROM TAPE TO DISK BASED EDITOR ASSEMBLER, THE BASIC LOADER OR SOURCE CODE LISTING MUST BE ENTERED AND SAVED TO DISK FIRST.

```
10 FOR I = 29184 TO 29290
20 READ A:POKE I,A:B=B+A: NEXT
30 IFB<>11582THENPRINT"ERROR":END
40 CLS:PRINT:PRINT
50 PRINT"INSERT DISK,CLOSE DOOR & PRESS  RETURN"
60 A$=INKEY$
70 IFINKEY$<>CHR$(13)THEN70
80 SOUND30,1
90 IF PEEK(16384) = 170THEN120
100 PRINT"NO DISK DRIVE TO USE"
110 END
120 BSAVE"TAPEDISK",7200,726A
130 END
140 DATA96,243,175,50,157,122,205,225,53,58,210,50,57,50,56,56
150 DATA33,96,56,205,4,56,221,33,35,120,205,104,56,56,51,229
160 DATA237,82,56,46,229,193,225,205,115,63,18,205,142,56,19
170 DATA11,121,176,32,243,205,117,55,221,190,0,32,22,205,117
180 DATA55,221,190,1,32,14,43,43,34,26,123,62,255,119,35,119
190 DATA251,195,0,123,33,74,56,251,205,167,40,243,24,170,33,224
200 DATA113,17,192,113,1,30,0,237,176,24,157,0,0
```

```
                          001 ;      TAPEDISK
                          002 ;LOADER FOR TAPE TO DISK
                          003 ;FOR EDITOR ASSEMBLER
                          004 ;VERSION 3  FEB 15 1989
                          005 ;ORIGIN 7200H
A5CC F3                   006       DI
A5CD AF                   007       XOR  A
A5CE 32 9D 7A             008       LD   (7A9DH),A
A5D1 CD E1 35             009 NST   CALL 35E1H
A5D4 3A D2 7A             010       LD   A,(7AD2H)
A5D7 FE F3                011       CP   0F3H
A5D9 20 4C                012       JR   NZ,FND
A5DB 21 60 38             013       LD   HL,3860H
A5DE CD 04 38             014       CALL 3804H
A5E1 DD 21 23 78          015       LD   IX,7823H
A5E5 CD 68 38             016       CALL 3868H
A5E8 38 33                017       JR   C,LDER
A5EA E5                   018       PUSH HL
A5EB ED 52                019       SBC  HL,DE
A5ED 38 2E                020       JR   C,LDER
A5EF E5                   021       PUSH HL
A5F0 C1                   022       POP  BC
A5F1 E1                   023       POP  HL
A5F2 CD 73 3F             024 LOOP  CALL 3F73H
A5F5 12                   025       LD   (DE),A
A5F6 CD 8E 38             026       CALL 388EH
A5F9 13                   027       INC  DE
A5FA 0B                   028       DEC  BC
A5FB 79                   029       LD   A,C
A5FC B0                   030       OR   B
A5FD 20 F3                031       JR   NZ,LOOP
A5FF CD 75 37             032       CALL 3775H
A602 DD BE 00             033       CP   (IX+00)
A605 20 16                034       JR   NZ,LDER
A607 CD 75 37             035       CALL 3775H
A60A DD BE 01             036       CP   (IX+01)
A60D 20 0E                037       JR   NZ,LDER
A60F 2B                   038       DEC  HL
A610 2B                   039       DEC  HL
A611 22 1A 7B             040       LD   (7B1AH),HL
A614 3E FF                041       LD   A,0FFH
A616 77                   042       LD   (HL),A
A617 23                   043       INC  HL
A618 77                   044       LD   (HL),A
A619 FB                   045       EI
A61A C3 00 7B             046       JP   7B00H
A61D 21 4A 38             047 LDER  LD   HL,384AH
A620 FB                   048       EI
A621 CD A7 28             049       CALL 28A7H
A624 F3                   050       DI
A625 18 AA                051       JR   NST
A627 21 E0 71             052 FND   LD   HL,71E0H
A62A 11 C0 71             053       LD   DE,71C0H
A62D 01 1E 00             054       LD   BC,30
A630 ED B0                055       LDIR
A632 18 9D                056       JR   NST
BYTES FREE :- 22342 ERRORS : 00000
```

ENHANCING THE LIST COMMAND 8
WRITTEN BY D.MITCHELL

I HAVE OFTEN NEEDED THE ADDRESS OF LINES IN A BASIC PROGRAM AND I USED A MONITOR TO DO THIS UNTIL I DISCOVERED THE LIST / LLIST ROUTINE CALLS AN ADDRESS IN RAM NOT ONLY DID THE ROUTINE DO THIS BUT IT WAS VERY HELPFULL IN THAT THE REGISTERS CONTAINED VITAL INFORMATION.

```
EG. :- HL  =  LINE NUMBER ADDRESS
       BC  =  NEXT LINE ADDRESS
       DE  =  MAX. LINE NUMBER SPECIFIED
```

FROM THIS I WAS ABLE TO WRITE THE TWO PROGRAMS BELOW BOTH OF WHICH PRINT THE START ADDRESS OF THE BASIC LINES.

LISTH PRINTS IN HEX AND LISTD PRINTS IN DECIMAL.

THE ADDRESSES THAT EITHER PROGRAM PRINTS WILL NOT BE THE SAME ADDRESS AS YOU WILL GET WITH LARRY TAYLOR'S FIND ROUTINE. DON'T GET ME WRONG THE FIND ROUTINE PRINTS THE ADDRESS OF THE FIRST COMMAND IN THE LINE WHERE LISTH AND LISTD PRINTS THE ADDRESS OF THE LINE NUMBER.

WHEN LISTING WITH EITHER PROGRAM DON'T TRY TO MODIFY ANY LINE WITHOUT FIRST REMOVING THE LINE ADDRESS.

EDITORS COMMENT - WHEN TYPING IN THE LISTINGS BELOW, DO NOT TYPE IN THE 5 DIGIT DECIMAL AND 4 DIGIT HEX NUMBERS AT START OF EACH LINE. I'VE INCLUDED THEM FOR EXAMPLE ONLY. LISTD & LISTH WORKS WITH BOTH THE LIST & LLIST COMMANDS.

** LISTING FOR LISTD **

```
31465 5 REM TO TURN FUNCTION OFF POKE 31199,201
31507 6 REM TO TURN FUNCTION ON POKE  31199,195
31549 10 FORI=31273TO31319:READA:POKEI,A:B=B+A:NEXT
31584 15 IFB<>5937THENPRINT"DATA ERROR":END
31613 16 POKE30862,41:POKE30863,122:X=USR(0)
31645 20 DATA33,58,122,34,224,121,62,195,50,223,121,205,201,1,195,25
31706 30 DATA26,197,213,245,229,126,35,110,101,111,43,223,48,12,225
31766 40 DATA229,43,43,205,175,15,62,32,205,42,3,225,241,209,193,201
```

** LISTING FOR LISTH **

```
7AE9 5 REM TO TURN FUNCTION OFF POKE 31199,201
7B13 6 REM TO TURN FUNCTION ON POKE  31199,195
7B3D 10 FORI=31273TO31363:READA:POKEI,A:B=B+A:NEXT
7B60 20 IFB<>11203THENPRINT"DATA ERROR":END
7B7E 25 POKE30862,41:POKE30863,122:X=USR(0)
7B9E 30 DATA33,58,122,34,224,121,62,195,50,223,121,205,201,1,195,25
7BD8 40 DATA26,197,213,245,229,126,35,110,101,111,43,223,48,12,225
7C17 50 DATA229,43,43,205,88,122,62,32,205,42,3,225,241,209,193,201
7C54 60 DATA124,205,97,122,125,205,97,122,201,79,203,63,203,63,203
7C90 70 DATA63,203,63,246,48,254,58,56,2,198,7,205,42,3,121,230,15
7CCC 80 DATA246,48,254,58,56,2,198,7,205,42,3,201
```

WRITTEN BY BOB KITCH

FOUR SUBROUTINES ARE PROVIDED THAT CAN BE CALLED FROM WITHIN ANY BASIC PROGRAM. OFTEN IT IS DESIRABLE TO BE ABLE TO PRESERVE A 2K HI-RES SCREEN SO THAT A LO-RES (TEXT) SCREEN CAN BE DISPLAYED. AFTER CERTAIN INSTRUCTIONS ARE CARRIED OUT, THE HI-RES SCREEN MAY NEED TO BE RESTORED. THE BASIC COMMANDS SET & RESET ARE MUCH TOO SLOW FOR THIS APPLICATION. THE FAST BLOCK MOVE COMMAND INHERENT IN THE Z80 IS MADE FOR THE TASK.

THE SUBROUTINES GIVEN ARE GENERAL AND RELOCATABLE FOR ANY VZ. THE BLOCK MOVE CONSTRUCT IS CALLED VIA THE USR COMMAND. THE MOVE ROUTINE IS MODIFIED DURING EXECUTION SO THAT IT IS USED AS A BLOCK FILL ROUTINE AND ALSO MOVES THE SCREEN IMAGE BOTH WAYS BETWEEN VIDEO RAM AND THE HI-MEM BUFFER.

1. SUBROUTINE 3000 - INITIALIZATION.

THIS SECTION SHOULD BE CALLED EARLY IN THE MAINLINE PROGRAM. IT LOWERS TOM, RESERVES 2400 BYTES OF HI-MEM, LOADS 14 BYTES OF MACHINE CODE AND SETS UP A NUMBER OF POINTERS. ENSURE THAT THESE VARIABLES ARE NOT RE-USED DURING PROGRAM EXECUTION.

MR & LR - MSB AND LSB OF STARTING ADDRESS FOR ROUTINE.
MS & LS - DITTO FOR STORAGE BUFFER.
MV & LV - DITTO FOR VIDEO RAM.
SS & SS+1 - ADDRESS FOR SOURCE.
DS & DS+1 - ADDRESS FOR DESTINATION.
ZS & ZS+1 - ADDRESS FOR LD (HL),XX BYTES.

THE 14 BYTES OF M/L ARE LOADED INTO THE BUFFER AND ARE FIRST SET UP AS A DESTRUCTIVE BLOCK FILL.

```
LD HL,SOURCE
LD DE,DESTINATION
LD BC,SIZE
LD (HL),CHARACTER
LDIR
RET
```

THE INITIAL COLOUR OF THE SCREEN BUFFER CAN BE SET BY ALTERING THE SECOND PIECE OF DATA LOADED IN LINE 3330. THE ADDRESSES FOR SOURCE AND DESTINATION ARE INITIALLY SET TO DUMMY VALUES OF 255D.

2. SUBROUTINE 4000 - CLEAR HI-MEM BUFFER.

THIS IS THE BLOCK FILL CALL WHICH IS ALSO NEEDED EARLY IN THE PROGRAM TO FILL THE 2K BUFFER WITH A CERTAIN CHARACTER. IT SETS UP THE USR VECTOR, POKES IN THE SOURCE AND DESTINATION ADDRESSES AND FINALLY MODIFIES THE M/L CODE TO A BLOCK MOVE TYPE.

```
LD HL,BUFFER      ;LS,MS
LD DE,BUFFER+1    ;L1,M1
LD BC,2048D
NOP
NOP
LDIR
RET
```

3. SUBROUTINE 4000 - MOVE SCREEN TO HI-MEM BUFFER.

THIS ROUTINE IS CALLED WHENEVER THE VIDEO SCREEN LOCATED FROM 7000H TO 77FFH IS TO BE MOVED INTO THE HI-MEM BUFFER. A LO-RES SCREEN CAN THEN BE DISPLAYED UNDER PROGRAM CONTROL.

```
LD HL,7000H    ;LV,MV
LD DE,BUFFER   ;LS,MS
LD BC,2048D
NOP
NOP
LDIR
RET
```

4. SUBROUTINE 5000 - MOVE HI-MEM BUFFER TO VIDEO RAM.

THIS SETS UP THE BLOCK MOVE AS SHOWN BELOW AND RESTORES THE HI-RES SCREEN THAT HAD PREVIOUSLY BEEN BUFFERED.

```
LD HL,BUFFER
LD DE,7000H
LD BC,2048D
NOP
NOP
LDIR
RET
```

WELL THAT'S ABOUT IT! GENERALLY SPEAKING, SUBROUTINES 3000 AND 4000 SHOULD BE RUN EARLY IN THE CALLING PROGRAM, AS THEY SET UP AND PRE-LOAD THE BUFFER AREA. I TRUST THAT USERS FIND THESE ROUTINES VERY USEFUL - THEY HAVE BEEN INVALUABLE WHEN WRITING GRAPHICS EDITORS.

AS A FOOTNOTE, IT IS WORTH RECORDING THE ACTION OF THE LDIR OPCODE. IT'S ACTION IS AS FOLLOWS :-

I/ ASSIGN (HL) TO (DE)
II/ INC HL
III/ INC DE
IV/ DEC BC
V/ REPEAT UNTIL BC=0

EDITORS COMMENT - WHEN TYPING IN THE BASIC LISTING DO NOT TYPE IN THE REM LINES AND COMMENTS AS THEY ARE THERE TO EXPLAIN PROGRAM OPERATION.

```
1 '*****************************
2 '***  SCREEN MODE MOVER    ***
3 '*** A SET OF SUBROUTINES ***
4 '***   TO SWITCH BETWEEN   ***
5 '***      HI & LO RES      ***
6 '***           BY          ***
7 '***       BOB KITCH       ***
8 '***          6/87         ***
9 '*****************************
10 '
```

```
15 'THIS SET OF SUBROUTINES IS USED IN BRUCE'S ART GALLERY.
2998 '
2999 '***INITIALIZATION - CALL AT START OF PROGRAM.
3000 M1=PEEK(30898):L1=PEEK(30897)     :'***FIND TOM.
3010 TM=M1*256+L1                      :'***TOM.
3020 TM=TM-2400                        :'***RESERVE 2.4K BUFFER.
3030 MS=INT(TM/256):LS=TM-MS*256       :'***NEW TOM.
3040 POKE 30898,MS:POKE 30897,LS       :'***SET NEW TOM.
3050 CLEAR 300                         :'***RESET POINTERS.
3060 TM=PEEK(30898)*256+PEEK(30898)    :'***NEW TOM.
3070 TM=TM+1                           :'***START ADDR. BUFFER.
3080 MR=INT(TM/256):LR=TM-MR*256       :'***START POINTERS.
3090 FOR MS=TM TO TM+13                :'***LOAD IN M/C.
3100    READ LS:ID=MS                  :'***SET UP ADDR.
3110    IF ID>32767 THEN ID=ID-65536   :'***CONV. TO S.INT.
3120    POKE ID,LS                     :'***POKE IN MOVE CODE.
3130 NEXT MS
3140 SS=TM+1                           :'***ADDR. SOURCE.
3150 IF SS>32767 THEN SS=SS-65536      :'***CONV. TO S.INT.
3160 DS=TM+4                           :'***ADDR. DESTIN.
3170 IF DS>32767 THEN DS=DS-65536      :'***CONV. TO S.INT.
3180 ZS=TM+9                           :'***ADDR. 2K BUFF.
3190 IF ZS>32767 THEN ZS=ZS-65536      :'***CONV. TO S.INT.
3200 TM=TM+30                          :'***START OF SCRN BUFF.
3210 MS=INT(TM/256):LS=TM-MS*256       :'***START OF 2K BUFF.
3220 MV=112:LV=0                       :'***START OF VIDEO RAM.
3230 RETURN
3298 '
3299 '***DESTRUCTIVE BLOCK FILL ROUTINE.
3300 DATA 033,255,255     :'LD HL,SOURCE.
3310 DATA 017,255,255     :'LD DE,DESTINATION
3320 DATA 001,000,008     :'LD BC,SIZE
3330 DATA 054,000         :'LD (HL),CHARACTER
3340 DATA 237,176         :'LDIR
3350 DATA 201             :'RET
3998 '
3999 '***CLEAR BUFFER ROUTINE.
4000 POKE 30863,MR:POKE 30862,LR       :'***SET USR() POINTERS.
4010 POKE SS,LS:POKE SS+1,MS           :'***SOURCE ADDR.
4020 M1=MS:L1=LS+1                     :'***CALC SOURCE+1.
4030 IF L1>255 THEN L1=0:M1=M1+1       :'***CHECK RANGE.
4040 POKE DS,L1:POKE DS+1,M1           :'***DESTINATION ADDR.
4050 S=USR(0)                          :'***SET 2K BUFFER.
4060 POKE ZS,0:POKE ZS+1,0             :'***INSERT NOP'S
4070 RETURN
4998 '
4999 '***MOVE 2K SCREEN BUFFER TO TOM BUFFER.
5000 POKE 30863,MR:POKE 30862,LR       :'***SET USR() POINTERS.
5010 POKE SS,LV:POKE SS+1,MV           :'***SET SOURCE ADDR.
5020 POKE DS,LS:POKE DS+1,MS           :'***SET DESTINATION ADDR.
5030 S=USR(0)                          :'***MOVE SCREEN TO HI-MEM.
5040 RETURN
5998 '
5999 '***MOVE TOM BUFFER INTO 2K VIDEO RAM...
6000 POKE 30863,MR:POKE 30862,LR       :'***SET USR() POINTERS.
6010 MODE(1)                           :'***HI-RES MODE.
6020 POKE SS,LS:POKE SS+1,MS           :'***SET SOURCE ADDR.
6030 POKE DS,LV:POKE DS+1,MV           :'***SET DESTINATION ADDR.
6040 S=USR(0)                          :'***MOVE SCREEN TO LO-MEM.
6050 RETURN
7000 END
```

```
PICT COPY MENU - MODE(0)
E - ERA  FILE   M - REN  FILE
L - LOAD PICT   S - SAVE PICT
D - DRIVE       P - SAVE PROG
F - DIRECTORY   B - MEM  PEEK
I - INIT DISK   G-GET   M-MENU
SELECT DESIRED FUNCTION
```

```
P-PAGE-1 ↑-INV  -MODE:0> Q-QUIT
A  AUSSFLAG
B  CHECKER
C  PICTMENU
D  PAGEMENU
X-LIST  Z-SHOW  T-GET   M-MENU
1-2 DRIVE - 1   C/R-COPY @-SAVE
```

```
10 GOTO45

45 POKE30897,204:POKE30898,191:REM"SET NEW TOP AT C000H

56 GOSUB 1000:GOTO 300

64 IFA$=" "THENPRINT@480,SP$;:RETURNELSERETURN

150 GOSUB70:PRINT@418,"LOAD PICTURE ";:SOUND30,1
151 INPUTN$:GOSUB60:AD=31555:GOSUB40
152 IFA$<>" "THEN300
154 MODE(HL):GOSUB18:SOUND25,1:GOSUB1200:GOTO550
158 :
160 GOSUB70:PRINT@418,"SAVE PICTURE ";:SOUND30,1
162 INPUTN$:GOSUB60:IFA$<>" "THEN300
164 GOSUB1100:IFHL=0 THEN AD=31609:GOSUB40:GOSUB22:GOTO 168
166 AD=31577:GOSUB40:GOSUB20
168 SOUND25,1:GOTO550

330 PRINT"  I - INIT DISK  G-GET  M-MENU":PRINTS$

362 IFD$="↑"THENGOSUB1100:GOTO550

410 PRINT"P-PAGE-"CHR$(241+PA)"  ↑-INV  -MODE(0)  Q-QUIT";

435 PRINT"X-LIST  T-GET  Z-SHOW  M-MENU";:RD=492
440 PRINT" 1-2 DRIVE -    C/R-COPY @-SAVE";

470 IFX$="%"THENCLS:LIST895-990

490 IFX$="↑"THENGOSUB1100:GOTO550
495 :
530 GOSUB 1200
992 :
994 REM"BLOCK MOVE ROUTINE
1000 X=31273:POKEX,33:POKEX+1,0:POKEX+2,112:POKEX+3,17:POKEX+4,0
1010 POKEX+5,192:POKEX+6,1:POKEX+7,0:POKEX+8,8:POKEX+9,237
1020 POKEX+10,176:POKEX+11,201:RETURN
1085 :
1090 REM"MOVE SCREEN TO 2 VIDEO RAM
1100 MODE(HL):SOUND25,1:POKE30862,41:POKE30863,122
1110 POKE31275,192:POKE31278,112:X=USR(X)
1120 POKE30862,80:POKE30863,52:RETURN
1185 :
1190 REM"MOVE SCREEN TO HI-MEM (C000H)
1200 POKE30862,41:POKE30863,122
1210 POKE31275,112:POKE31278,192:X=USR(X)
1220 POKE30862,80:POKE30863,52:RETURN
```

FRONT COVER - WHEN PICTCOPY IS RUN AND BEFORE ANY SCREEN IS LOADED FROM DISK, TRY THE GET (^) KEY. ON MY VZ 300 IT PRODUCES A CHECKER BOARD PATTERN WHICH IS REPRODUCED ON FRONT COVER. ANYBODY HAVE ANY IDEAS ON WHY THIS HAPPENS ?

MY THANKS TO ROBERT QUINN FOR INCORPORATING A BLOCK MOVE ROUTINE IN PICTCOPY FOR ME WHICH CONSISTS OF 12 BYTES WHICH ARE POKED IN INDIVIDUALLY IN LINES 1000-1020. THIS CLEVER APPROACH DOES AWAY WITH NEED FOR DATA STATEMENTS.

ROBERT ALSO POINTED OUT THAT I HAD SOME ERRORS IN PICTCOPY, ISSUE # 21, PAGE 13 WHICH HE CORRECTED.

ERROR 1) - UNRESOLVED GOSUBS IN LINE 60 WHENEVER QUIT OPTION WAS USED. THAT WAS A BIT OF SLOPPY PROGRAMMING ON MY PART.

ERROR 2) - LINE 85 SHOULD NOT BE THERE AS IT APPEARED WHEN I MERGED ROUTINES TOGETHER AND MISSED IT COMPLETELY TILL EAGLE EYED ROBERT BROUGHT IT TO MY ATTENTION.

THE ADDITION OF BLOCK MOVE MAKES PICTCOPY VERY VERSATILE ALLOWING FULL USE OF DISK FUNCTIONS WHILE RETAINING HI/LO-RES SCREEN IN MEMORY AND RECALLED JUST BY A KEY PRESS OR TWO.

I PRESUME YOU HAVE ENTERED DISK UTILITY FROM ISSUE # 21 AND ADDED PICT COPY FROM ISSUE # 22 TO WHICH YOU MUST ADD THE EXTRA LINES SHOWN ON PREVIOUS PAGE. SOME OF THE NEW LINES WILL OVERWRITE EXISTING LINES WITH SOME CHANGES TO BOTH MENUES. AND NOW TO THE ALTERATIONS/ADDITIONS AND NEW FUNCTIONS.

*) - INV - PRESSING (*) KEY WILL TOGGLE BETWEEN NORMAL AND INVERSE SCREEN.

NOTE - UPON LOADING A HI/LO-RES SCREEN FROM EITHER MENU IT WILL BE AUTOMATICALLY MOVED INTO HI-MEMORY 2K BUFFER FOR LATER RETRIEVAL. SLIDE SHOW SCREENS DO NOT GO INTO BUFFER.

^) - GET - THE (^) UP ARROW KEY WILL GET HI/LO-RES SCREEN FROM HI-MEM AND PUT IT INTO 2K VIDEO RAM WHERE IT CAN BE VIEWED, SAVED OR PRINTED OUT.

IF THE SCREEN IS DISPLAYED IN WRONG MODE, THEN USE 'Q' TO RETURN TO MENU AND PRESS MINUS (-) KEY TO SELECT CORRECT MODE. PRESSING (^) GET KEY WILL DISPLAY SCREEN IN CORRECT MODE.

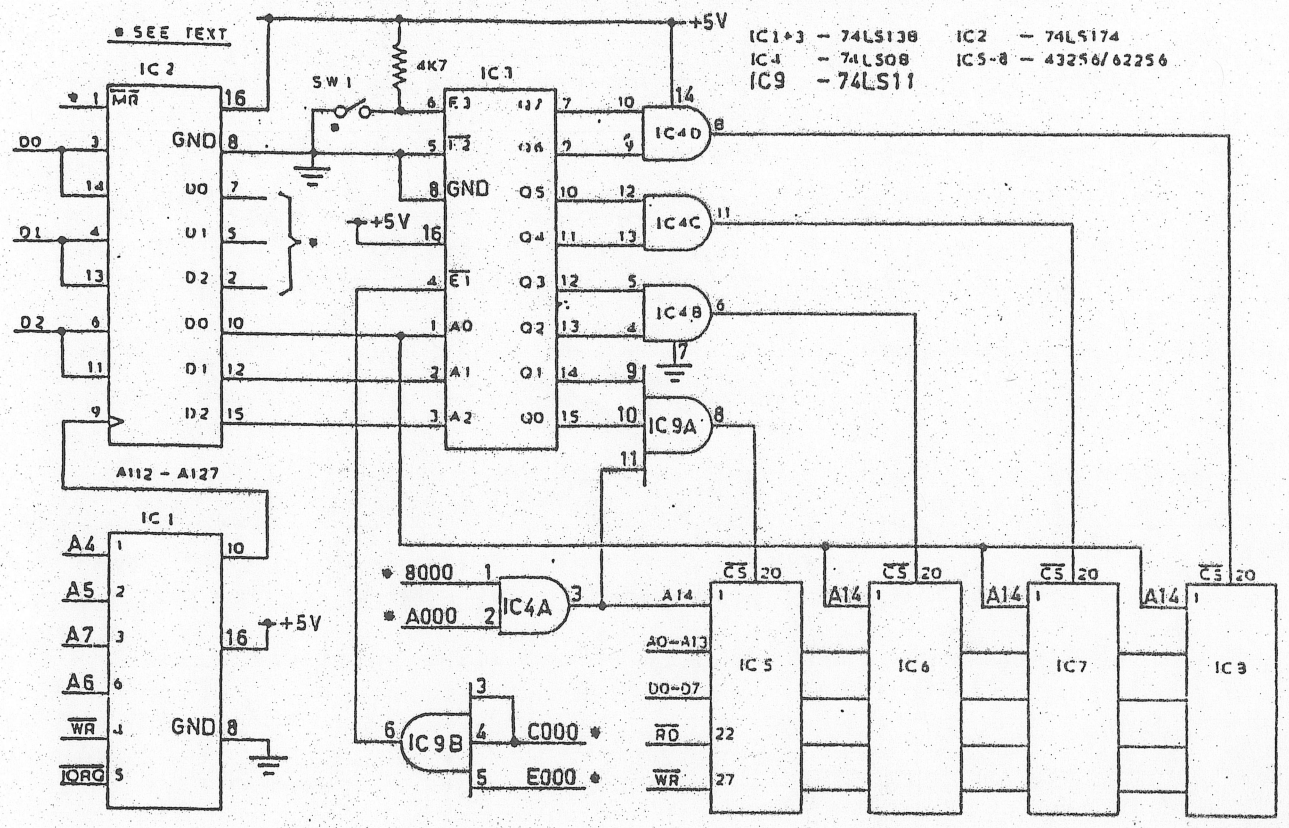
S) - SAVE PICT - THIS FUNCTION WILL SAVE A HI/LO-RES SCREEN TO DISK USING FILENAME OF YOUR CHOICE. THE SCREEN CAN BE LOADED IN PREVIOUSLY FROM EITHER MENU, BUT NOT FROM SLIDE SHOW.

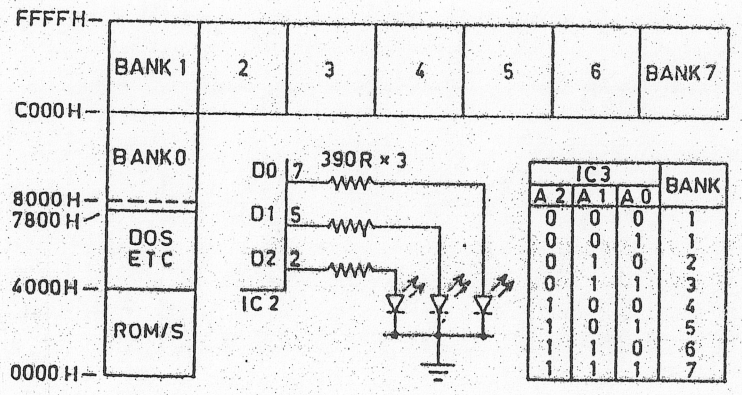
THIS PROGRAM IS A GOOD EXAMPLE OF USING ROUTINES AS BUILDING BLOCKS TO DESIGN A PROGRAM WITH DESIRED FUNCTIONS.

OFFER - AS THIS PROGRAM IS SPREAD OVER SEVERAL ISSUES AND IS A MERGING OF SEVERAL UTILITIES SOME READERS MAY HAVE PROBLEMS GETTING IT TO WORK PROPERLY. I CAN PROVIDE YOU WITH THE PROGRAM ON DISK TOGETHER WITH SOME HI-RES SCREENS.

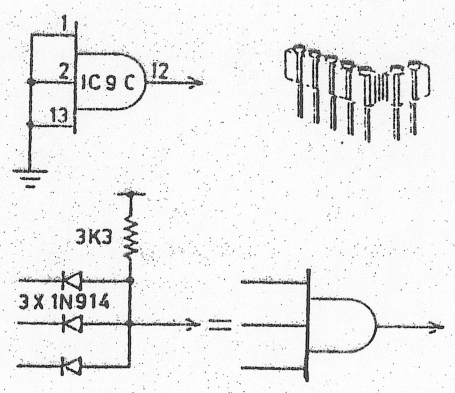
COST - $4.00 - INCLUDES DISK PLUS POST & PACKING.
CONTACT JOE LEON 22 DRURY STREET WALLSEND N.S.W. 2287

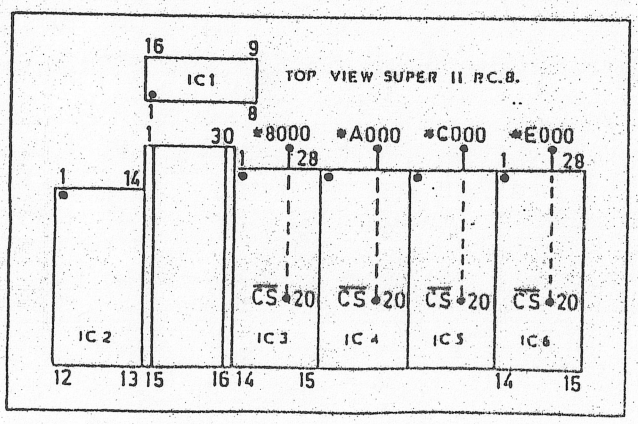
VZ 200/300 128K SIDEWAYS RAM (C) JOSEPH P. LEON 1988





IC3			BANK
A2	A1	A0	
0	0	0	1
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7





VZ 200
6K/34K RAM BOARD

TOP VIEW					BOTTOM VIEW				
A15	1	30	MREQ		30	1	A15		
A13	2	29	A12		29	2	A13		
A14	3	28	A11		28	3	A14		
+5V	4	27	A7		27	4	+5		
A8	5	26	A6		26	5	A8		
A9	6	25	A5		25	6	A9		
$\overline{WR}$	7	24	A4		24	7	$\overline{WR}$		
$\overline{RD}$	8	23	A3		23	8	$\overline{RD}$		
A10	9	22	A2		22	9	A10		
$\overline{78XX}$	10	21	A1		21	10	$\overline{78XX}$		
D7	11	20	A0		20	11	D7		
D6	12	19	D0		19	12	D6		
D5	13	18	D1		18	13	D5		
D4	14	17	D2		17	14	D4		
D3	15	16	GND		16	15	D3		

PLEASE READ DISCLAIMER ON BOTTOM OF PAGE 3 BEFORE ATTEMPTING PROJECT.

AS PROMISED IN LAST ISSUE, THE CHANGES PRESENTED MAKE 128K S/WAYS RAM COMPATIBLE WITH 64K RAM PACK. BEACAUSE THIS ARTICLE IS THIRD IN SERIES SPREAD OVER MANY MONTHS I DECIDED TO PRESENT MODIFIED CIRCUIT DIAGRAM COMPLETE SO NEW READERS WOULD'NT MISS OUT, ALSO PREVIOUS ERRORS HAVE BEEN FIXED UP.

WE'LL START BY HAVING A LOOK AT MODIFIED MEMORY MAP. YOU'LL NOTE BANK ZERO IS NOW AT 8000H-BFFFH AND IS FIXED AT THIS LOCATION, SAME AS 64K RAM PACK. BANKS 1-7 OCCUPY C000H-FFFFH RANGE, BUT ONLY ONE BANK AT A TIME.

UPON POWER UP OR RESET ALL O/PUTS ON IC 2 ARE FORCED LO WHICH IN TURN PRESENT 3 LO'S AT A0-A2 ON IC 3 ADDRESS INPUTS. THIS WILL MAKE Q0 LO AND Q1-Q7 HI. THE O/P'S OF IC 3 ARE TAKEN TO 3 X 2 I/P & 1 X 3 I/P AND GATES.

ONLY ONE O/P ON IC 3 CAN BE LO AT ANY ONE TIME AND A LO ON ANY AND GATE I/P WILL MAKE IT'S O/P LO THEREBY ENABLING ONE OFF THE FOUR 32K RAM CHIPS. ADDRESS 14 I/P ON THE 32K RAM CHIPS IS USED TO SELECT TOP OR BOTTOM 16K. WHEN EITHER *8000H OR *A000H GO LO, IC 4A'S O/P GOES LO ENABLING IC 5 AND ALSO SELECTING BOTTOM 16K (BANK 0) VIA ADDRESS 14 I/P ON IC 5.

NOW WHEN EITHER *C000H OR *E000H GO LO, IC 9A'S O/P GOES LO IN TURN ENABLING IC 3 WHICH IN TURN ENABLES ONE OF THE 32K RAM CHIPS DEPENDING ON BANK SELECTED. AT POWER UP OR RESET BANK 1 IS ENABLED. THE 8K CS (CHIP SELECT) SIGNALS ARE GENERATED BY A 74LS138 DECODER ON 34K SUPER II RAM BOARD, AND LIKE IC 3, ONLY ONE OF IT'S O/P'S CAN BE LO AT ANY ONE TIME.

SWITCH 1 IS NEEDED IF 128K S/WAYS RAM IS TO BE USED WITH WORDPRO CARTRIDGE. IT DISABLES ALL BANKS EXEPT FOR BANK ZERO.

THE OUT COMMAND IS USED TO SWITCH TO DESIRED BANK, IE. - OUT 112,1 TO 7

LIKE THE 64K RAM PACK, SELECTING BANK ZERO WILL IN REALITY SELECT BANK 1. BANK ZERO HAS BEEN TAKEN OUT OF THE SELECTION PROCESS USING THE OUT COMMAND VIA IC 9A & IC 4A. REFER TO THE MODIFIED TRUTH TABLE.

CONSTRUCTION USING ETI 687 34K RAM BOARD :-

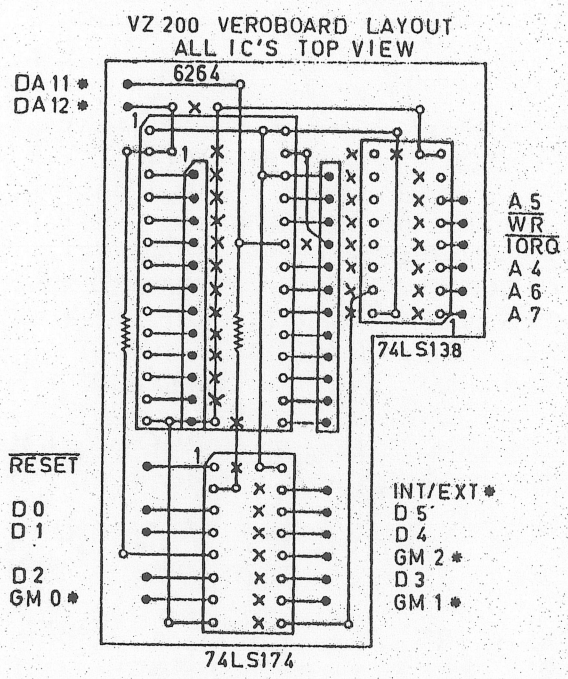
IC 1 ON 34K RAM BOARD GENERATES 8K BLOCK CS SIGNALS WHICH ARE MARKED BY ASTERICKS ON PCB LAYOUT. THEY ARE CONNECTED TO IC 4A & IC 9A. THE O/P'S FROM IC 4B, 4C, 4D & IC 9A ARE TAKEN TO RESPECTIVE CS SIGNALS ON 34K RAM BOARD. OF COURSE THE LINK WIRES ARE NOT USED ANY MORE. 32K RAM CHIPS HAVE TWO ADDITIONAL ADDRESS LINES OVER 8K CHIPS WHICH ARE A13 & A14. BEND ALL A13, PINS 26 & A14, PINS 1 UP 90 DEG. BEFORE INSERTING INTO SOCKETS.

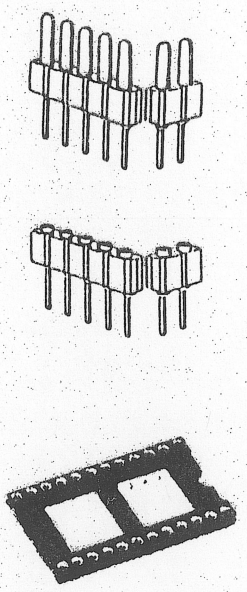
CONNECT ALL PINS 26 TOGETHER USING LINK WIRE AND CONNECT TO A13 WHICH IS SHOWN ON PINOUT BESIDES THE 34K RAM BOARD LAYOUT. A14 PINS ON 32K RAM CHIPS ARE CONNECTED AS SHOWN ON CIRCUIT AND NOT TO A14 ON ADDRESS BUS.

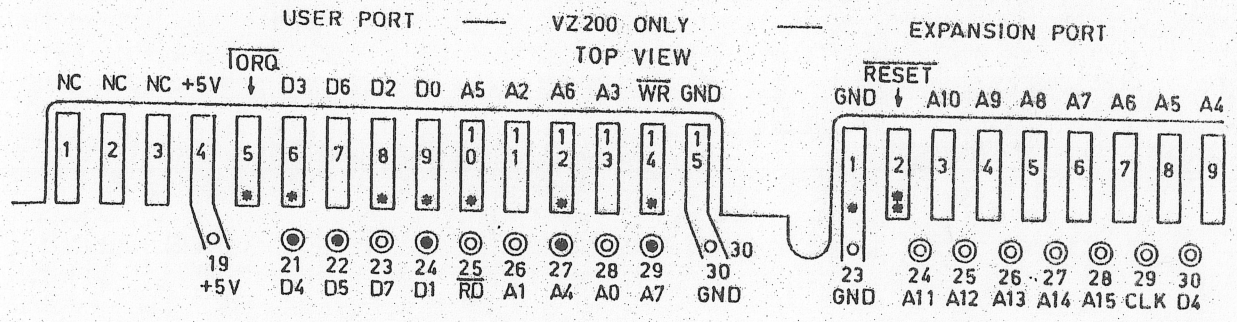
FIVE IC'S ARE USED FOR CONTROLLING CIRCUITRY AND COULD BE CONSTRUCTED ON PIECE OF VEROBOARD. THE REQUIRED SIGNALS COULD BE TAKEN FROM THE 30 PIN SOCKET ON 34K RAM BOARD OR FROM USER PORT WHOSE PINOUT APPEARS ON PAGE 16. IF USING USER PORT THEN MACHINE INSERT SOCKET STRIPS COULD BE SOLDERED TO IT. SINGLE STRAND SHIELDED WIRE CAN THEN BE USED TO PLUG INTO STRIP WITHOUT SOLDERING AND IN CASE OF WIRING ERRORS EASILY RECTIFIED.

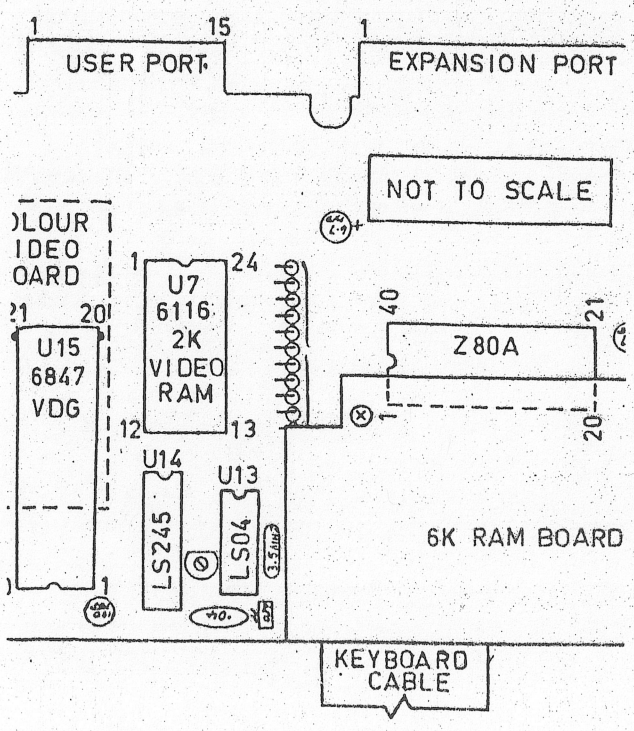
NOTE - YOU CAN START WITH ONE 32K RAM CHIP AND ADD OTHERS AS NEED ARISES, BUT YOU HAVE TO PLACE THEM IN THE FOLLOWING ORDER - IC 5, 6, 7 & 8.

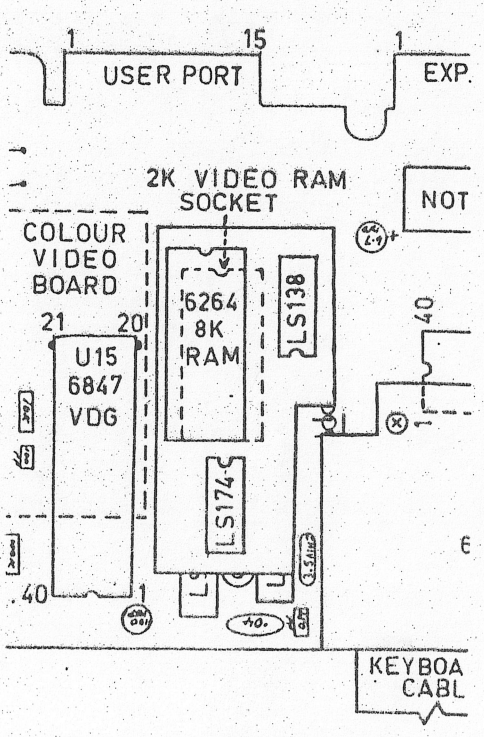
NOTE FOR DISK USERS - ANY ATTEMT TO USE BANKS 2 TO 7 WILL RESET THE DOS BECAUSE OF THE DOS COMMUNICATION REGION AT TOP OF BANK 1.


VZ 200 VEROBOARD LAYOUT
ALL IC'S TOP VIEW
6264
DA 11 *
DA 12 *
A 5
WR
IORQ
A 4
A 6
A 7
74LS138
RESET
D 0
D 1
D 2
GM 0 *
INT/EXT *
D 5
D 4
GM 2 *
D 3
GM 1 *
74LS174




USER PORT — VZ200 ONLY — EXPANSION PORT
TOP VIEW
NC NC NC +5V IORQ D3 D6 D2 D0 A5 A2 A6 A3 WR GND
19 +5V
21 D4
22 D5
23 D7
24 D1
25 RD
26 A1
27 A4
28 A0
29 A7
30 GND
RESET
GND A10 A9 A8 A7 A6 A5 A4
23 GND
24 A11
25 A12
26 A13
27 A14
28 A15
29 CLK
30 D4


USER PORT
EXPANSION PORT
NOT TO SCALE
U7
6116
2K
VIDEO
RAM
U15
6847
VDG
Z80A
U14
LS245
U13
LS04
6K RAM BOARD
KEYBOARD
CABLE


USER PORT
2K VIDEO RAM
SOCKET
COLOUR
VIDEO
BOARD
6264
8K
RAM
LS138
U15
6847
VDG
LS174

PLEASE READ DISCLAIMER ON BOTTOM OFF PAGE 3 BEFORE ATTEMPTING THIS PROJECT.

NOTE - VZ SUPER GRAPHICS ISSUE # 22 - THE PARTS LIST FOR SUPER GRAPHICS WAS ACCIDENTALLY LEFT OUT AND IS REPRODUCED HERE FOR BOTH VZ 200 & VZ 300.

1 X 74LS138 1 X 74LS174 1 X 6264 8K RAM 2 X 6K8 RESISTORS
1 X 0.1 CERAMIC OR MONOLITHIC CAPACITOR
1 X 24 PIN M/I (MACHINE INSERT) SOCKET
1 X 32-50 PIN M/I SOCKET STRIP (VZ 200).
1 X 32 PIN M/I ADAPTOR STRIP
1 X 40 PIN M/I SOCKET (VZ 300)
1 X 40 WAY IDC FLAT RIBBON CABLE DIL PLUG (VZ 300).
1 X MINIATURE PUSH BUTTON SWITCH
19 X 15CM (6") LENGHTS OF INSULATED HOOK UP WIRE
1 PIECE OF VEROBOARD 17 HOLES ACCROSS BY 19 TRACKS DOWN (VZ 300).
1 PIECE OF VEROBOARD 16 HOLES ACCROSS BY 25 TRACKS DOWN (VZ 200).

NOTE - ON PAGE 16, TOP RIGHT ARE DEPICTED M/I (MACHINE INSERT) ADAPTOR STRIP, M/I SOCKET STRIP & 24 PIN M/I SOCKET.

PREPARING VZ 200 - FIRST TAKE VZ CASE APART AND REMOVE CIRCUIT BOARD FROM CASE. NEXT REMOVE RF SHIELD BY UNSOLDERING IT FROM PCB.

6116 - UNSOLDER 2K VIDEO RAM OR CUT PINS ON ONE SIDE AND LEVER BACK AND FORTH TILL PINS BREAK ON OTHER SIDE. REMOVE PIN STUBS AND INSPECT BOARD FOR DAMAGE AND REPAIR IF NEEDED. NEXT SOLDER A 24 PIN M/I SOCKET IN 6116'S PLACE.

6847 (VDG) - REMOVE TWO SCREWS ON COLOUR VIDEO BOARD AND LIFT BOARD UP OUT OF ROAD SO YOU HAVE ACCESS TO 6847. USING SMALL SIDECUTTERS CUT THE FOLLOWING PINS AS CLOSE TO THE PCB AS POSSIBLE AND BEND UP 90 DEG. CUT PINS 27, 29 & 30 ON 6847 AND BEND UP 90 DEG. DO NOT FORGET TO REMOVE PIN STUBS FROM PCB.

PREPARING VZ 200 PLUG IN MODULE :-

VEROBOARD - THE TRACKS RUN HORIZONTALLY, FROM LEFT TO RIGHT. OPEN AND SOLID CIRCLES DENOTE USED HOLES ON VEROBOARD. VERTICAL LINES DENOTE INSULATED LINK WIRES. HORIZONTAL LINES SIMPLY DENOTE USED TRACKS AND ARE SHOWN FOR CLARITY ONLY. THE (X'S) DENOTE CUT TRACKS.

USING AN 1/8" (3MM) DRILL BIT CUT TRACKS MARKED WITH AN (X) CHECKING AND DOUBLE CHECKING ALL ARE CORRECT AND HAVE'NT MISSED ANY. NEXT ISERT ALL LINK WIRES. PLEASE NOTE LINK WIRE UNDER 6264, PIN 26 HAVE TWO WIRE LINK ENDS GOING INTO SAME HOLE.

MOUNT THE RESISTOR ON THE LEFT NEXT. THE RESISTOR SHOWN UNDER 6264 IC IS MOUNTED UNDER THE BOARD WITH ONE END SHARING HOLE WITH LINK WIRE. BE CAREFULL NOT TO SHORT OUT TRACKS UNDERNEATH.

M/I ADAPTOR STRIP - THIS IS A DOUBLE SIDED SINGLE ROW OF PINS WHICH CAN BE SNAPPED TO DESIRED LENGTH. ONE SIDE HAS THINNER PINS THAN OTHER SIDE. SNAP TWO LENGTHS OF 12 PINS EACH. INSERT THE STRIPS IN THE 24 PIN RAM SOCKET WITH THINNER PINS AT BOTTOM.

NEXT PUT VEROBOARD ON TOP OF PINS AND LOWER DOWN ON THEM. LIFT V/BOARD UP TILL IT JUST CLEARS COMPONENTS ON PCB AND MARK HEIGTH ON PINS. REMOVE PINS FROM SOCKET AND SOLDER THICK PINS TO V/BOARD TO MARKED HEIGTH AND CUT ALL PINS ON LEFT STRIP FLUSH WITH TOP OF V/BOARD.

THE RIGHT ADAPTOR STRIP NEED NOT BE CUT FLUSH AS A LINK WIRE IS SOLDERED TO ONE PIN. THE TWO STRIPS ON V/BOARD LAYOUT MARKED WITH SOLID CIRCLES IS WHERE THE ADAPTOR STRIPS ARE SOLDERED TO.

BEFORE PROCEEDING FURTHER TRY PLUGGING IN MODULE INTO 2K VIDEO RAM SOCKET TO CHECK IF IT PLUGS IN PROPERLY AND CLEARS ALL COMPONENTS AND REMOVE. USING NO SOCKETS SOLDER THE 74LS174, 74LS138 & 6264 IC'S IN NEXT MAKING SURE THE IC'S ARE SOLDERED IN RIGHT WAY ROUND. USING THE V/BOARD LAYOUT AS A GUIDE SOLDER ALL 19 OFF BOARD LINK WIRES.

USING BIT OF THIN CARDBOARD LIKE FROM BACK OF WRITING PAD CUT A PIECE BIT LARGER THAN V/BOARD. CUT TWO SLOTS FOR THE TWO ADAPTOR STRIPS TO GO THROUGH AND INSERT C/BOARD UNDER BOARD WHICH IS USED TO INSULATE BOTTOM OF V/BOARD FROM COMPONENTS IN VZ.

PLUG IN MODULE IN VACANT 2K VIDEO RAM SOCKET AND SOLDER THE 5 WIRES TO 6847 FIRST WHICH ARE DENOTED BY ASTERICKS. PINS 20 & 21 ON PCB ARE NOT CONNECTED TO ANYTHING AND FOR THAT REASON THEY DO NOT HAVE TO BE CUT AND CAN BE SOLDERED TO DIRECT.

THE REST OF THE WIRES CAN BE SOLDERED DIRECTLY TO FINGERS ON USER PORT. IF YOU CAN GET SINGLE STRAND SHIELDED WIRE THEN YOU COULD SOLDER ONE END TO MODULE AND SOLDER M/I SOCKET STRIPS TO USER PORT AND PLUGGING OTHER END INTO STRIP. THIS APPROACH SAVES A LOT OF SOLDERING AND IN CASE OF WIRING ERRORS ARE SIMPLY RECTIFIED. THE USER PORT PINOUT IS MARKED BY ASTERICKS FOR SIGNALS NEEDED BY MODULE.

THE SAME APPROACH CAN BE USED ON 6847 BY SOLDERING M/I SOCKET STRIP/S TO PINS 20, 21, 27, 29 & 30 AND PLUGGING WIRES INTO THEM.

NOTE - IF YOUR VZ 200 HAS AN ETI 687 34K RAM BOARD INSTALLED THEN ONE CORNER OFF IT'S PCB WILL HAVE TO BE CUT AWAY SO MODULE CAN BE PLUGGED IN.

RESET BUTTON - SOLDER 2 WIDE M/I SOCKET TO PINS 1 & 2 ON EXPANSION PORT AND THEN SOLDER TWO LENGTHS OF WIRE TO PB SWITCH AND OTHER ENDS TO ANOTHER 2 WIDE M/I SOCKET. THIS GIVES YOU A PLUG IN RESET BUTTON. YOU COULD MOUNT RESET BUTTON ON BOTTOM RIGHT LIP OF CASE ABOVE POWER SWITCH.

IF YOUR WORK CHECKS OUT OK THEN REASSEMBLE VZ. THE RF SHIELD COULD BE LEFT OFF UNLESS YOU LIVE NEAR A POWER STATION. IN THAT CASE CUT HOLE IN SHIELD IF MODULE TOO HIGH AND RESOLDER TO PCB.

WHENEVER YOU POWER UP OR RESET THE VZ TYPE IN OUT 32,8 AND PRESS RETURN OR YOU'LL END UP WITH ONLY 1K (1024 BYTES) FOR HI-RES SCREENS. IT'S A GOOD IDEA TO INCLUDE AN OUT 32,8 AT START IN ALL YOUR PROGRAMS.

TESTING - REFER TO ISSUE # 22

CAPTURING HI/LO-RES SCREENS :-

WITH SUPER GRAPHICS AND RESET BUTTON INSTALLED IN YOUR VZ IT BECOMES A SIMPLE MATTER TO CAPTURE ANY HI/LO-RES SCREEN FROM ANY PROGRAM. SIMPLY LOAD YOUR PROGRAM INTO MIDDLE PAGE USING FOLLOWING METHOD :-

```
OUT32,25:MODE(1):BRUN"INVADERS"
```

WHAT WE HAVE DONE IS TO SELECT PAGE 1 (MIDDLE PAGE), GRAPHICS MODE(6) AND THEN RUN PROGRAM. IT'S IMPORTANT TO DO IT AS DESCRIBED ABOVE BECAUSE ONCE YOU SELECT PAGE 1 YOU CAN'T SEE WHAT YOU'RE ENTERING ON SCREEN. ONCE YOUR PROGRAM HAS LOADED IT WILL BE DISPLAYED IN PAGE 1, BUT WON'T LOOK NORMAL.

PRESS RESET BUTTON WHEN DESIRED SCREEN APPEARS. THE VZ WILL BE RESET AND AS WILL BE VIDEO RAM TO PAGE 0. DESIRED SCREEN WILL BE SAFE AND SOUND IN PAGE ONE. IN NEXT ISSUE I'LL DESCRIBE HOW IT CAN BE RETRIEVED FROM PAGE 1.

1 X CASIO FX-702 P PROGRAMMABLE (IN BASIC) CALCULATOR
1 X CASIO FA-2 CASSETTE INTERFACE
1 X CASIO FP-10 PRINTER & 8 ROLLS OF PAPER

CALCULATOR ONLY ON SECOND SET OF BATTERIES AND PRINTER USED ONLY 2 ROLLS OF PAPER. CASSETTE INTERFACE USED ONLY FOR SHORT TIME AS BOUGHT VZ WITH PRINTER, ETC. ALL HAND BOOKS ARE INCLUDED AS SUPPLIED.

PRICE - $150.00 PLUS POSTAGE - WILL NOT SEPERATE

CONTACT NEVILLE HUGHES AT 4 BINYA ST. WHITTON 2705
OR PHONE (069) 55 2719

VZ USER - NO WORD AS YET IF ANYONE IS TAKING OVER ITS PUBLICATON

LE'VZ OOP - VZ MAGAZINE - VSOFTWAREZ/SOFTWARE/HARDWARE FOR SALE
J.C.E. D'ALTON 39 AGNES ST. TOOWONG QUEENSLAND 4066

VZ DOWN UNDER - VZ MAGAZINE - 6 ISSUES - $18.00 PER YEAR
HARRY HUGGINS 12 THOMAS ST. MITCHAM 3132 VICTORIA

VZ-LINK - PETER J. HILL P.O.BOX 1972 C.P.O. AUCKLAND N.Z.

WAVZ - GRAEME BYWATER P.O. BOX 388, MORLEY W.A. 6062

BRISBANE VZ USERS WORKSHOP - C/O 63 TINGALPA ST. WYNUM WEST 4178
SOFTWARE FOR SALE - DISK MENU

HUNTER VALLEY VZ USERS' GROUP - P.O. BOX 161 JESMOND N.S.W. 2299
SECRETARY/EDITOR-JOE LEON (049) 51 2756
PRESIDENT-ROSS WOODS (049) 71 2843
TREASURER - MATTHEW TAYLOR -**- COMMITTEE MEMBER - COLIN BRIDGE

SUBSCRIPTION - H.V.VZ.JOURNAL - 6 MONTHS $9.00 - 12 MONTHS $18.00
NEW ZEALAND - 6 MONTHS $12.00 - 12 MONTHS $24.00

MEETINGS - FIRST FRIDAY OF MONTH AT JESMOND NEIGHBOURHOOD CENTRE
MORDUE PARADE - REAR STOCKLAND MALL (BIG W) JESMOND

NOTE :- WHEN WRITING TO ANY ABOVE OR H.V.VZ. USERS' GROUP FOR INFORMATION PLEASE ENCLOSE A S.S.A.E. OR NZ 2 INT. REPLY COUPONS.

VZ SCREEN ED - $25.00 + $2.00 POST & HANDLING.
VZ SCREEN ED IS A NEW DRAWING PROGRAM FOR HI-RES SCREENS WITH MANY POWERFUL OPTIONS. BOTH TAPE AND DISK USERS ARE CATERED FOR.

SEND YOUR CHEQUE OR MONEY ORDER TO :-
MATTHEW TAYLOR SOFTWARE -- OR PHONE (049) 75 2350
38 FISHING POINT ROAD RATHMINES NSW 2283 AUSTRALIA

YOU CAN SEE MATTHEW AT CLUB MEETINGS FOR A DEMONSTRATION OR PURCHASE OF SCREEN ED.

E & F WORD PROCESSOR PATCH 3.3

THIS LATEST PATCH WRITTEN BY DAVE MITCHELL WILL CONVERT YOUR E & F TAPE WORD PROCESSOR FOR FULL DISK USE WHILE RETAINING ALL TAPE FUNCTIONS. BELOW ARE THE DISK COMMANDS & ADDED FUNCTIONS :-

LOAD, SAVE, ERASE, RENAME, DIRECTORY, INITIALIZE, UPDATE, DRIVE 1 & 2, SHIFTLOCK & IMBEDDED PRINTER CONTROL CODES

PATCH 3.3 HAS PROVISION FOR IMBEDDING PRINTER CONTROL CODES IN TEXT AND FAST SAVING AND LOADING OF TEXT DATA TO AND FROM DISK USING BLOCK SAVE/LOAD TECHNIQUES.

MINIMUM MEMORY REQUIREMENTS :-
VZ 300 + 16K RAM PACK - VZ 200 + 26K

PATCH 3.3 IS COPYRIGHT TO AND ONLY AVAILABLE FROM :-
HUNTER VALLEY VZ USERS' GROUP P.O.BOX 161 JESMOND 2299
N.S.W. AUSTRALIA - Phone (049) 51 2756

PRICE - AUS/NZ AU$20.00 - UPDATE - AUS-$10.00 - NZ-AU$11.00.
UPDATING AVAILABLE ONLY TO PREVIOUS PURCHASERS OF OUR PATCHES.

FOR MORE INFORMATION WRITE TO H.V.VZ.U.G. ENCLOSING A SSAE.

FOR PRIVATE SALE

EXTENDED DOS V1.3 -$15.00

THE PREVIOUS VERSION HAS BEEN UPDATED WITH EXTRA COMMANDS ADDED.

OLD COMMANDS - MERGE, DIRA, LDIRA, DIRB, LDIRB, OLD, OLD., DEC, HEX, STATUSA AND LSTATUSA. STATUSA AND LSTATUSA ALSO WORKS WITH VERSION 1.0 DOS.

NEW COMMANDS :-
MENU - LOADS AND RUNS BINARY OR TEXT MENU PROGRAM FROM DISK.
CODE - SIMPLIFIES USING PRINTER CONTROL CODES DIRECTLY OR FROM WITHIN A PROGRAM.
LTAB - IS FOR SETTING OF LEFT MARGIN.
MOVE - MOVES BASIC FILE FROM DISK TO CHOSEN MEMORY ADDRESS.
UPD - ERASES OLD FILE AND SAVES WITH SAME FILE NAME.

MENU/FILE COPIER/DISK ORGANIZER - $15.00

THIS UTILITY WILL READ YOUR DISK DIRECTORY AND PRESENT YOU WITH SEVERAL OPTIONS. USING THE CURSOR YOU CAN RUN/BRUN ANY PROGRAM OR SELECT FILE COPY, REN, ERASE, DRIVE 1 OR 2, ETC. BESIDES COPYING TEXT AND BINARY FILES ALL OTHER FILES CAN BE COPIED AS WELL EXEPT FOR DATA FILES.

FOR PURCHASE OR INFO CONTACT - DAVE MITCHELL - (079) 27 8519
24 ELPHINSTONE St. NORTH ROCKHAMPTON QUEENSLAND 4701

FOR INFORMATION OR DEMONSTRATION IN NEWCASTLE AREA CONTACT :-
JOE LEON - (049) 51 2756 - 22 DRURY ST. WALLSEND NSW 2287